Strategies Comparison for Game Theoretic Cyber Situational Awareness and Impact Assessment

Dan Shen, Genshe Chen, and Leonard Haynes
Intelligent Automation, Inc.
Rockville, MD 20855
{dshen, gchen, lhaynes}@i-a-i.com

Erik Blasch
AFRL/SNAA,
WPAFB, OH 45433
erik.blasch@wpafb.af.mil

Abstract - This paper compares different defense strategies against various attacks utilizing a dynamic game theoretic data fusion framework for cyber network defense. In our game theoretic framework, Alerts generated by Intrusion Detection Sensors (IDSs) or Intrusion Prevention Sensors (IPSs) are fed into the data refinement (Level 0) and object assessment (L1) data fusion components. High-level situation/threat assessment (L2/L3) data fusion based on Markov game model and Hierarchical Entity Aggregation (HEA) are proposed to refine the primitive prediction generated by adaptive feature/pattern recognition and capture new unknown features. A Markov (Stochastic) game method is used to estimate the belief of each possible cyber attack pattern. Game theory captures the nature of cyber conflicts: determination of the attacking-force strategies is tightly coupled to determination of the defense-force strategies and vice versa. A software tool is developed to demonstrate and compare the performance of different defense strategies used in game theoretic high level information fusion for cyber network defense situations and a simulation example shows the enhanced understating of cyber-network defense.

Keywords: Cyber Defense, Situation Awareness, Impact assessment, Information Fusion, Game Theory, Networks Security.

1 Introduction

There are increasing needs for research in the area of cyber situational awareness which includes data transfer, storage, and recovery security [15]. The protection and defense against cyber attacks to computer networks is becoming inadequate as the hacker knowledge sophisticates and as the network and each computer system become more complex. When evaluating the security of a network, it is rarely enough to consider the presence of isolated vulnerabilities [18]. Large networks typically contain multiple platforms and software packages and employ several modes of connectivity between various types of intruders from internal (i.e. espionage) to external (i.e. terrorists) disgruntled people. Inevitably, such networks have security holes that escape notice of even the most diligent system administrators.

Cyber attacks in the past were generally one-dimensional, mainly in the form of denial of service (DoS) attacks, computer viruses or worms, or unauthorized intrusions (hacking). These attacks were mainly launched against websites, mail servers, or client machines. Recently, attacks have fundamentally changed – cyber threats are undergoing a diversification that is resulting in multi-stage and multi-dimensional attacks that utilize and/or target a variety of attack tools and technologies [16, 17]. Most contemporary attacks, the latest generation of worms for instance, make use of a variety of different exploits, propagation methods, and payloads. Infected machines may be used to launch attacks against other targets or their data could be accessed or deleted. Even more worrisome, the trend is toward an intensification of this development, potentially resulting in the emergence of many more sophisticated cyber attacks.

Therefore, cyberspace security requires next-generation network management and intrusion detection systems that combine both short-term sensor information and long-term knowledge databases to provide decision-support systems and cyberspace command and control. Recent advances in applying data fusion techniques to cyber situational awareness are promising. Some pioneering works focused on high-level descriptions of these approaches are presented in [9-10]. Significant results of cyber situation awareness are achieved, but the assessment of the impact of a cyber attack and the prediction of an attacker's intent, or high level data fusion, are not fully explored. Stochastic game theory was introduced to meet the challenge [14].

Game theory is not a new concept in cyber defense. Current game theoretic approaches [1-3] for cyber network intrusion detection and decision support are based on static matrix games and simple extensive games, which are usually solved by game trees. However, these matrix game models lack the sophistication to study multi-players with relatively large actions spaces, and large planning horizons. Some partial results of our dynamic Markov game approach have been published in [14].

In this paper, we extend our game theoretic data fusion with data mining framework for cyber defense to include

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
JUL 2007		**00-00-2007 to 00-00-2007**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Strategies Comparison for Game Theoretic Cyber Situational Awareness and Impact Assessment	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Air Force Research Laboratory,AFRL/SNAA,Wright Patterson AFB,OH,45433**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
10th International Conference on Information Fusion, 9-12 July 2007, Quebec, Canada.

14. ABSTRACT
see report

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**8**	

the human in the loop. Particularly, we study various defense strategies based on different game equilibrium solutions such as pure Nash strategies, mixed Nash strategies, and Mini-max strategies.

The framework itself can be applied to multiple domains from urban warfare to cyber attack. In this framework, a decentralized Markov game is used to model the evolution of Enemy Course of Actions (ECOAs) originating from an initial prediction generated by pattern recognition. Our approach has several features:

1) *Recognition/Refinement/Learning Structure.* If the observed features are already associated with adversary intents, we can easily obtain them by pattern recognition. In some time-critical applications, the primitive prediction can be used before it is refined by relatively time-consuming high-level data fusion. Unknown or new cyber attack patterns are input to dynamic learning module to update the feature-situation association database;

2) *Decentralized.* Each cluster or team of IDSs makes decisions mostly based on the local information. We put more autonomies in each group allowing for more flexibility;

3) *Markov Decision Process (MDP)* can effectively model uncertainties in the cyber network environment;

4) *Game framework* is an effective and ideal model to capture the nature of network conflicts: the determination of one side's strategies is tightly coupled to that of the other side's strategies and vice versa;

5) *Neutral players*: white (neutral) objects (normal network nodes) are modeled as one of the multi-players so that their possible COA will be estimated and considered by the other players;

The rest of the paper is organized as follows. Section 2 describes our proposed framework. Section 3 presents a Markov model for cyber network. Section 4 describes the simulation tool and experimental results. Section 5 concludes the paper.

2 Framework for Cyber Situation Awareness

We propose an information fusion based decision and control framework (Fig. 1) to detect and predict the multistage stealthy cyber attacks. Our cyberspace security system has two fully coupled major parts: 1) *Data fusion module* (to refine primitive awareness and assessment, and to identify new cyber attacks); and 2) *Dynamic/adaptive feature recognition module* (to generate primitive estimations, and to learn new identified new or unknown cyber attacks).

Various logs and alerts generated by Intrusion Detection Sensors (IDSs) or Intrusion Prevention Sensors (IPSs) are fed into the L1 data fusion components. The fused objects and related pedigree information are used by a feature/pattern recognition module to generate primitive prediction of intents of cyber attackers. If the observed features are already associated with adversary intents, we can easily obtain them by pattern recognition. In some time-critical applications, the primitive prediction can be used before it is refined; because the high-level data fusion refinement operation is relatively time-consuming in the multiplicative of probability calculations.

High-level (L2 and L3) data fusion based on Markov game models is proposed to refine the primitive prediction generated in stage 1 (Dynamic and Adaptive Feature Recognition) and capture new or unknown cyber attacks. The Markov (Stochastic) game method (MGM) is used to estimate the belief of each possible cyber attack graph. Game theory can capture the nature of cyber conflicts: the determination of the attacking-force strategies is tightly coupled to the determination of the defense-force strategies and vice versa. Also MGM can deal with the uncertainty and incompleteness of the available information. We propose a graphical model to represent the structure and evolution of the above-mentioned Markov game models so that we can efficiently solve the graphical game problem.

The captured unknown or new cyber attack patterns will be associated to related L1 results in the *dynamic learning block*, which takes deception reasoning, trend/variation identification, and distribution models and calculations into account. Our approach to deception detection is heavily based on the application of pattern recognition techniques to detect and diagnose what we call out-of-

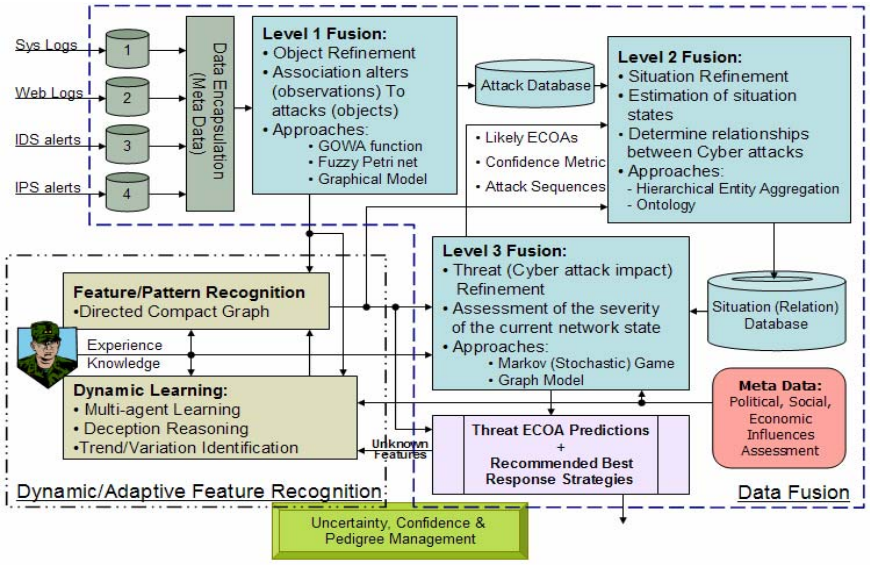

Fig. 1: A Data Fusion Approach for Cyber Situation Awareness and Impact Assessment

normal (anomaly) conditions in the cyber environment. The results of dynamic learning or refinement shall also be used to enhance L2 and L3 data fusion. This adaptive process may be considered as level 4 data fusion (process refinement; see the 2004 DFIG model [11]).

In this paper, we will focus on the Level 3 data fusion part in the overall framework shown in Fig. 1. We will show how concepts from dynamic Markov game can be used to find and evaluate strategies for defending cyber network attacks.

3 Markov Game Model

To address the cyber network security problem from a system control and decision perspective, we present a Markov game model [4]. In general, a Markov (stochastic) game is specified by (i) a finite set of players N, (ii) a set of states S, (iii) for every player $i \in N$, a finite set of available actions D^i (we denote the overall action space $D = \times_{i \in N} D^i$), (iv) a transition rule $q: S \times D \to \Delta(S)$, (where $\Delta(S)$ is the space of all probability distributions over S), and (v) a payoff function $r: S \times D \to R^N$. For the cyber decision support and attacker intent inference problem, we obtain the following distributed discrete time Markov game (we revise the Markov game model [5] used for battle-space and focus on the cyber network attack domain properties):

Players (Decision Makers) --- Cyber attackers, network defense system, and normal network users are players of this Markov game model. We denote cyber attackers as the red team, network defense systems (IDSs, Firewalls, Email-Filters, Encryption) as the blue team, and a normal network user as white team. The cooperation within the same team is also modeled so that the coordinated cyber network attacks can be captured and predicted.

State Space --- All the possible states of involved network nodes consist of the state space. For example, the web-server (IP = 26.134.3.125) is controlled by attackers. To determine the optimal IDS deployment, we include the defense status for each network nodes in the state space. So for the i^{th} network node, there is a state vector $s^i(k)$ at time k.

$$s^i(k) = (f, p, a)^T \quad (1)$$

where f is the working status of the i^{th} network node, p is the protection status, T is the transpose operator, and a is the status of being attacked. "Normal" and "malfunction" are typical values of f with the meaning that the node is in the normal working status or malfunction (Recall that in battle space cases, the function status of any unit values can be "undestroyed", "damaged", or "destroyed"). p can be the defense unit/service (such as firewall, IDS and filter, with probability) assigned to the node and p = NULL means that the i^{th} node is unprotected. a is the status of being attacked. The type of attacks will be specified in Action Space.

Remark 1: It is not difficult to understand that the system states are determined by two factors: 1) previous states and 2) the current actions. So the whole system can be model by a first-order Markov decision process.

The overall system state at time k is

$$s_k = [s^1(k), s^2(k), \cdots, s^M(k)] \quad (2)$$

where M is the number of nodes in the involved cyber network.

Action Space --- At every time step, each player chooses targets with associated actions based on its local network information. For normal network users, the action types are http services, email services, ftp services, etc. The action-decision control of the i^{th} white player at time k is

$$u_W^i(k) = (t, v)^T \quad (3)$$

where vector t is the network node providing services and v is the service type requested. (We assume that the normal users know the server/service in advance). [Note: we use u for action decisions because action decisions typically are determined as utility functions with the higher payoff]

For red team (cyber network attackers), we consider the following types of *network-based attacks*:

- *Buffer overflow* (web attack): it occurs when a program does not check to make sure the data it is putting into a space will actually fit into that space. Vulnerability exists in Microsoft IIS 5.0 running on Windows 2000 that allows a remote intruder to run arbitrary codes on the victim machine, allowing them to gain complete administrative control of the machine. Apache HTTP Server version 1.3.19 could allow a remote attacker to send an HTTP request to cause the server to crash with unexpected behavior.
- *Semantic URL attack* (web attack): In semantic URL attack, a client manually adjusts the parameters of its request by maintaining the URL's syntax but altering its semantic meaning. This attack is primarily used against CGI driven websites. A similar attack involving web browser cookies is commonly referred to as cookie poisoning.
- *E-mail Bombing* (email attack): In Internet usage, an e-mail bomb is a form of net abuse consisting of sending huge volumes of e-mail to an address in an attempt to overflow the mailbox or overwhelm the server. The possible re-action is to identify the source of the email bomb/spam and configure your router (or have your Network Service Provider configure the router) to prevent incoming packets from that address.
- *E-mail spam* (email attack): Spamming is the abuse of electronic messaging systems to send unsolicited, undesired bulk messages. Spammers often collect addresses of prospective recipients from use-net

postings or from web pages, obtain them from databases, or simply guess them by using common names and domains. By popular definition, spam occurs without the permission of the recipients.

- *MALware attachment* (email attack): Malware is software designed to infiltrate or damage a computer system without the owner's informed consent. It is a blend of the words "malicious" and "software". Common MALware attacks are worms, viruses, trojan horses, etc.
- *Denial-of-service* (network attack): Denial-of-service (DoS) attack is an attempt to make a computer resource unavailable to its intended users. Typically the targets are high-profile web servers where the attack is aiming to cause the hosted web pages to be unavailable on the Internet. A distributed denial of service attack (DDoS) occurs when multiple compromised systems flood the bandwidth or resources of a targeted system, usually a web server(s). These systems are compromised by attackers using a variety of methods.

Remark 2: Some attacks may be multi-stage. For example, e-mail spam and MALware are used first to gain control of several temporal network nodes, which are usually not-well protected servers. Then DoS attack will be triggered to a specified and ultimate target. Our dynamic Markov game model can handle these attacks from a planning perspective. Our mixed Nash strategy pair is based on a fixed finite planning horizon. See Strategies for details.

For the blue team (network defense system), we consider the following *defense actions*:

- *IDS deployment*: we assume that there are limited IDSs. IDS deployment is similar to resource allocation (target selection) problems in traditional battle-space situations. We try to find an optimal deployment strategy to maximize the chance of detecting all possible cyber network intrusions.
- *Firewall configuration*: A firewall is an information technology (IT) security device which is configured to permit, deny or proxy data connections set and configured by the organization's security policy. Firewalls can either be hardware and/or software based.
- *Email-filter configuration*: Email filtering is the processing of e-mail to organize it according to specified criteria. Most often this refers to the automatic processing of incoming messages, but the term also applies to the intervention of human intelligence in addition to artificial intelligence, and to outgoing emails as well as those being received. Email filtering software inputs email and for its output, it might (a) pass the message through unchanged for delivery to the user's mailbox, (b) redirect the message for delivery elsewhere, or (c) throw the message away. Some e-mail filters are able to edit messages during processing.
- *Shut down or reset servers*.

Transition Rule --- The objective of the transition rule is to calculate the probability distribution over the state space $q(s_{k+1}|s_k,u_k^B,u_k^R,u_k^W)$, where s_k,s_{k+1} are system states at time k and $k+1$ respectively, u_k^B,u_k^R,u_k^W are the overall decisions of the blue team (network defense system), the red team (cyber attackers) and the white team (normal network users), respectively, at time step k. How to decide the overall actions for each team are specified in Strategies.

For each network node (server or workstation), the state of time k+1 is determined by three things: 1) state at time k; 2) control strategies of the three teams; and 3) the attack/defense efficiency. If we compare part 3) to battle-space domain, the efficiency is the analogue of kill probability of weapons.

For example, if the state of node 1 at time k is ["normal", "NULL", "NULL"], one component of the red action is "email-bombing node 1", one component of blue action is "email-filter –configuration-no-block for node 1", and all white actions are not related to node 1, then the probability distribution of all possible next states of node 1 is: ["normal", "email-filter-configuration", "email-bombing"] with probability 0.4; ["slow response", "email-filter-configuration", "email-bombing"] with probability 0.3; and ["crashed", "email-filter-configuration", "email-bombing"] with probability 0.3. The actual probabilities depend on the efficiency of attacking and defending actions.

Payoff Functions --- In our proposed decentralized Markov game model, there are two levels of payoff functions for each team (red, blue, or white): lower (cooperative within each team) level and higher (non-cooperative between teams) level payoff functions. This hierarchical structure is important to model the coordinated cyber network attacks and specify optimal coordinated network defense strategies and IDS deployment.

The lower level payoff functions are used by each team (blue, red or white side) to determine the cooperative team actions for each team member based on the available local information. For the j^{th} unit of blue force, the payoff function at time k is defined as $\phi_j^B\left(\tilde{s}_j^B(k),u_j^B(k),W^B(k);k\right)$, where $\tilde{s}_j^B(k)\subseteq s_k$ is the local information obtained by the j^{th} blue member, $u_j^B(k)$ is the action taken by the blue team member at time k, and $W^B(k)$, the weights for all possible action-target couples of blue force, is announced to all blue team members and determined according to the top level payoff functions from a team-optimal perspective.

$$\phi_j^B\left(\tilde{s}_j^B(k),u_j^B(k),W^B(k);k\right) = U\left(\tilde{s}_j^B(k)\right) - w\left(W^B(k),u_j^B(k)\right)C\left(u_j^B(k)\right) \quad (4)$$

where, $U\left(\tilde{s}_j^B(k)\right)$ is the utility or payoff of the current local network state. Usually, it is a negative value if a network node is in malfunction status due to a cyber attack. The specific value depends on the value of the network node. The counterpart in the battle-space domain is the target value of each platform. Function $w\left(W^B(k),u_j^B(k)\right)$ will calculate the weight for any specified action decision for the j^{th} member of the blue team based on the received $W^B(k)$, which is determined on a team level and indicates the preference and trend of team defense strategies. $C\left(u_j^B(k)\right)$ is the cost of action to be taken by the blue team member.

Similarly, we obtain the lower level payoff functions for the j^{th} member of red and white team,

$$\phi_j^R\left(\tilde{s}_j^R(k),u_j^R(k),W^R(k);k\right) = U\left(\tilde{s}_j^R(k)\right) - w\left(W^R(k),u_j^R(k)\right)C\left(u_j^R(k)\right) \quad (5)$$

$$\phi_j^W\left(\tilde{s}_j^W(k),u_j^W(k),W^W(k);k\right) = U\left(\tilde{s}_j^W(k)\right) - w\left(W^W(k),u_j^W(k)\right)C\left(u_j^W(k)\right) \quad (6)$$

Remark 3: It is well known that non-neutral civilians often play an active role in wars. That is, they are not just passively static but might purposefully take actions to help one side in a battle to minimize their losses or achieve some political purpose. Unfortunately, existing game theoretic models usually do not consider this situation, although collateral damage has been considered in a paper on a two-player game model by Cruz *et al* [6]. In this research, a three-player dynamic game model is formulated, in which two opposing forces and one normal player that might be either neutral or slightly biased [7]. In our current implementation, the white units only care about their possible losses. For an example, when a slower or malfunctioned network node is detected, normal network users will find a COA to keep themselves as far as possible from the node. In addition, there may be no cooperation between the white team members, so we can simply set $W^W(k)$ to 1.

Remark 4: In some instances of the use of game theory for military applications by others [1-3], it is almost always the case that zero-sum game theory is used. In zero-sum game theory, the players have opposite objectives. If one player maximizes an objective function, the other automatically minimizes it. This is equivalent to a player maximizing an objective function and the other player maximizing the negative of the same function. Since the sum of the objective functions is zero, the game is called a zero-sum game. But for the cyber network attack scenario, we propose a non-zero-sum game model for the following two reasons: 1) there are three players as mentioned in Remark 4; 2) even in the case without a white player, there are some cases the objective of attacking side and defense side are not opposite of each other. For example, the hackers may be deterred from any attacking actions by well-protected defense systems. In this case, payoffs of both sides decrease, which is conflicting the zero-sum assumption. So we model the cyber network attack and defense system as a non-zero-sum dynamic Markov game.

The top level payoff functions at time k are used to evaluate the overall performance of each team.

$$V^B(\tilde{s}^B(k),u_k^B;k)=\sum_{j=1}^{M^B}\phi_j^B\left(\tilde{s}_j^B(k),u_j^B(k),W^B(k);k\right) \quad (7)$$

$$V^R(\tilde{s}^R(k),u_k^R;k)=\sum_{j=1}^{M^R}\phi_j^R\left(\tilde{s}_j^R(k),u_j^R(k),W^R(k);k\right) \quad (8)$$

$$V^W(\tilde{s}^W(k),u_k^W;k)=\sum_{j=1}^{M^W}\phi_j^W\left(\tilde{s}_j^W(k),u_j^W(k),W^W(k);k\right) \quad (9)$$

In our approach, the lower lever payoffs are calculated distributively by each team member and sent back to network administrator via communication networks.

Strategies --- In this paper, we have tried several well known types of strategies. Here we only give a brief description about the following three of them.

Min-max strategies: This kind of strategies will give a conservative solution to minimize the possible maximum "loss". Actually, in our problem, it is a max-min solution in the sense that each player maximizes the possible minimum his payoffs. So, this kind of strategies is also called safest solutions, in which we consider the worst-case attacks from network threats.

Pure Nash Strategies: The Nash equilibrium (named after John Nash [8] who proposed it) is a kind of optimal collective strategy in a game involving two or more players, where no player has anything to gain by changing only his or her own strategy. If each player has chosen a strategy and no player can benefit by changing his or her strategy while the other players keep theirs unchanged, then the current set of strategy choices and the corresponding payoffs constitute a Nash equilibrium.

Mixed Nash Strategies: A mixed strategy is used in game theory to describe a strategy comprised of possible actions and an associated probability, which corresponds to how frequently the action is chosen. Mixed strategy Nash equilibria (NE) are equilibria where at least one player is playing a mixed strategy. It was proved by Nash that that every finite game has a Nash equilibria but not all has a pure strategy Nash equilibrium. While computing his

mixed NE strategy, each player pays attention only to the average payoff functions.

In our cyber network security application, mixed Nash strategies are preferred since the existence is guaranteed. In addition, the stochastic property of mixed Nash strategy is compatible to the Markov (stochastic) game model. Playing a mixed strategy can also keep your opponent off balance. The worst case payoff of a mixed strategy may be better than the worst case payoff of a pure strategy.

In our proposed approach, the solution to the Markov game is obtained via a *K* time-step look-ahead approach, in which we only optimize the solution in the *K* time-step horizon. *K* usually takes 2, 3, 4, or 5. The suboptimal technique is used successfully for reasoning in games such as chess, backgammon and monopoly.

Remark 5: The *K*-step look-ahead (or moving window) approach well fits the situations in which multi-step cyber network attacks occurs since we evaluate the performance of each team based on the sum of payoffs during a period of *K*-time steps.

Random defense strategies: Cyber network defense strategies are randomly generated. Each possible element in the Blue Action space is chosen with equal probability.

Probability-based defense strategies: A Markov Decision Process (MDP) is created to assist the cyber network defender to specify a defense strategy, which can maximize the cumulative function of the rewards.

4 Experiments

4.1 Simulation Tool

To evaluate our game theoretic approach for cyber attack prediction and mitigation and compare different strategies, we have constructed a Cyber Game Simulation Platform (CGSP) based on an open-source network experiment specification and visualization tool kit (ESVT). [ESVT] Through this event-based, interactive and visual simulation environment, various attack strategies (single stage or multi-staged) and scenarios can be easily played out and the effect of game theoretic attack prediction and mitigation can be visually and quantitatively evaluated. Fig.2 is a snapshot of the CGSP environment.

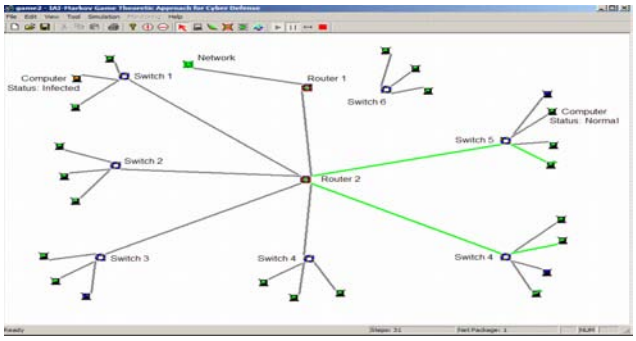

Fig.2 Cyber Game Simulation Platform (CGSP)

The implemented network components in this platform includes Computer (host), Switch, OSPF Router or Firewall, Link (connection), and (Sub) Network (Simulated by a node).

Besides the ordinary network properties such as processing capacity, bandwidth, and delay etc., CGSP components can be assigned a number of network attack containment or traffic mitigation properties to act as various defense roles, including smart IDS (intrusion detection systems), incoming traffic block, and outgoing traffic block. Additionally and more importantly, these defense roles or network defense properties can be deployed and re-deployed on the fly during a game simulation run-time based on the local intelligence and orders from higher-level command centers.

The color of a link represents the traffic volume on that link (in KBps and in Mbps). Light Gray: less than 1 percent of bandwidth. Green: more than 1 percent of bandwidth. Yellow: between green and red. Red: more than 30 percent of bandwidth

The color of a host indicates the host status. Red: Infected node. Green: Vulnerable node but not infected. Gray: Non-vulnerable node.

Some features of our tool include: (1) An *integrated environment* to plan and specify interactive network simulation experiments. At the first step, it can be used to draw the network topology and specify component properties such as susceptibility, server, or non-server, etc. Users can also change the properties of a group of components or all components by invoking the global component and script property configuration window. The topology builder is designed to be scalable and can support large topologies with thousands of components. (2) *Network protocol neutral*: Network packets and their communication are simulated by high-level event objects and their movement across various components. Specific network protocol details are ignored. We ignore many such technological details so we can focus on attack scenario construction and defense strategies. Preliminary emulation experiments have shown that such network protocol neutral simulation yields a relatively high fidelity.

(3) *Event driven simulation*. It means to represent and organize network dynamics by events in a network environment and trace all the events. A running event may trigger a new event to be generated. So every step of simulation is to let all the events to be finished if they can finish. Event driven is complemented by object driven since some host will generate new events even when there is no triggering event on the host (For example, during a worm break or a infected host participating in a DoS attack). We define a step of network simulation, which means a non-interruptible time period of the simulation. (4) For router simulation, if we use Dijkstra's Algorithm [12], then we have to generate a table for every router. Dijkstra's single-source shortest-path algorithm computes all shortest paths from a single vertex. But the storage of such table in every node wastes valuable computer resources. Another algorithm is called Floyd's all-pairs shortest-path algorithm, or the Floyd-Warshall All-Pairs-Shortest-Path algorithm. It solves all the shortest paths in the same step. We use this algorithm to get the "shortest path" [13].

In our simulation software, network attacks and defenses are simulated in CGSP by events. Live network packets and other communications are represented and simulated by the main network event queue. Users or software agents can inject packets or network events through the timed event (M/M/1) queue. Security alerts or logs are generated and stored in the security log queue.

There are a number of cyber attacks that are included in the CGSP implementation: Port scan, Buffer attack (to gain control), Data bomb or Email bomb from and to a single host, Distributed Denial of service from multiple hosts, Worm attack, and Root right hack (confidentiality loss). [Note: Both buffer attack victims and worm infectives will join the distributed denial of service when they receive the DDOS command.]

The arsenal of network defense team includes: Smart IDS (Accuracy and false positive adjustable), Directional traffic block (outgoing or incoming), Host Shutdown, Host Reset (shutdown and restart). [Note: Both SHUTDOWN and RESET will clear the infection status on the host.]

4.2 Computer Simulation and Experiments

In the simulated scenario as shown in Fig. 2, there are 23 computers, 2 routers, 7 switches, and 1 network. In this scenario, we first limit the look-ahead steps *K* to 2 (which means the defense side does not consider the multi-stage attacking patterns). In this case, we implemented Nash strategies for cyber network defense side. We can see from Fig. 3 and Fig. 4 that a target computer (web server) is infected or hacked. Then the computer (web server) will be used by attacking force to infect other more important target computers such as file servers or email servers. This two-step attacking scheme is based on two facts: 1) a public web server is easy to attack and 2) an infected internal computer (web server in this case) is more efficient and stealthy than an external computer to attack well protected computers such as data servers or email servers.

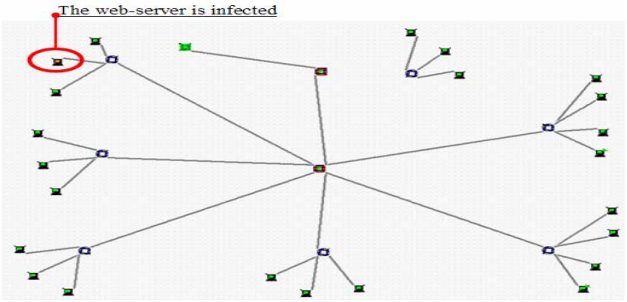

Fig. 3: A public web server is infected or hacked

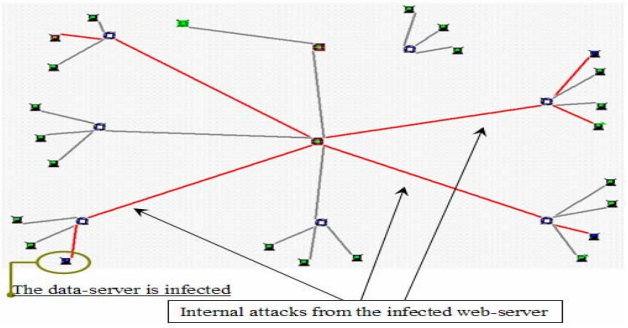

Fig. 4 Three more important data servers are attacked by the infected internal web server

In the next run, we set the look-ahead step *K*=5. Then no network nodes are infected or hacked during the simulation of 2 hours. If a public server is infected, the defense side can foresee the enemy's next attacking internal server from the infected network node. Then a shut-down or reboot action will be taken to destroy the multi-stage attack at the first stage.

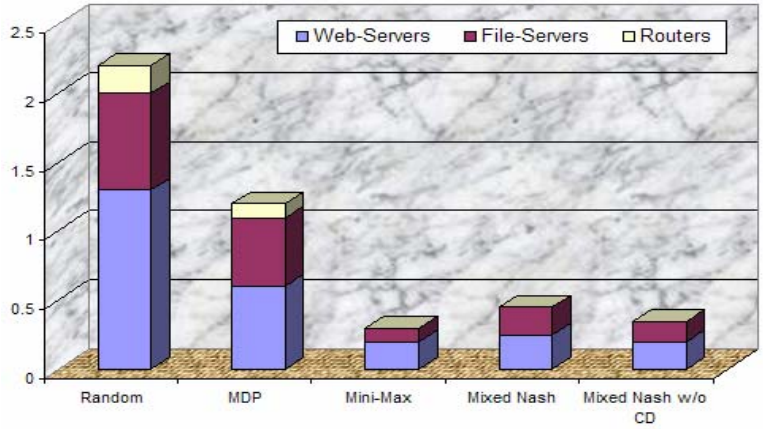

Fig. 5: Infection comparison of various options

In addition to the explained run, we performed many experiments. We compared the results using the various strategies, such as random defense strategies, probability-based defense strategies (from Markov decision process without consideration the interactions between defender and attacker), mini-max defense strategies, mixed Nash strategies (applied in the explained simulation as shown in Fig. 3 – Fig. 4), mixed Nash strategies without collateral damage consideration in the cost function of blue side. Since the simulation is stochastic, the results consist of the mean of 10 runs for each case with K=5, which are shown in Fig. 5 (Only the damage information of the Blue side is shown).

From the damage comparison results, we can see that our proposed Markov game framework with mixed Nash equilibrium for threat detection and situation awareness is better than the other methods if we consider the much higher defense costs in min-max defense strategies and the higher traffic volume in the Mixed Nash strategies without the collateral damage (CD).

5 Conclusion

In this paper, we have compared different defense strategies implemented in an information-fusion/data-mining based situation awareness and adversary intent inference in a cyber attack and network defense scenario. The network security system was evaluated and protected from a perspective of data fusion and system control. The goal of our approach was to examine the estimation of network states and projection of attack activities (similar to ECOA in the warfare scenario). We used Markov game theory's ability to "step ahead" to infer possible adversary attack patterns. Extensive simulations were performed to verify and illustrate the benefits of this cyber information fusion model. The performance of our algorithm was very promising and demonstrates the effective control tradeoffs associated with cyber-security information management based on cyber situational and threat aware information fusion.

Acknowledgement

The authors thank Mr. George Tadda and John Salerno from AFRL for their high degree of technical involvement in the project and support throughout the effort.

References

[1] K. Sallhammar, S. J. Knapskog and B. E. Helvik, "Using Stochastic Game Theory to compute the expected Behavior of attackers", *Proceedings, 2005 Symposium on Applications and the Internet Workshops*, 2005.

[2] T. Alpcan and T. Basar, "A game theoretic application to decision and analysis in Network Intrusion Detection", *42nd IEEE CDC 2003*, pp. 2595-2600, Maui, Hawaii, USA

[3] A. Agah, S. K. Das and K. Basu, "A non-cooperative game approach for intrusion detection in sensor networks", *Vehicular Technology Conference*, 2004. VTC2004-Fall. pp. 2902 – 2906

[4] L. S. Shapley, "Stochastic games," in *Proceedings of the National Academy of Sciences of the United States of America*, vol. 39, pp. 1095-1100, 1953.

[5] G. Chen, D. Shen, C. Kwan, J. B. Cruz, Jr., and M. Kruger, "Game Theoretic Approach to Threat Prediction and Situation Awareness," in *Proceedings of the 9th International Conference on Information Fusion*, Florence, Italy, July, 2006.

[6] J. Cruz, G. Chen, D. Garagic, X.Tan, D. Li, D. Shen, M. Wei, & X. Wang, "Team Dynamics and Tactics for Mission Planning," Proceedings, *IEEE Conference on Decision and Control*, December 2003.

[7] M. Wei, G. Chen, J. B. Cruz, C. Kwan, and M. Kruger, "Game theoretic modeling and control of military air operations with civilian players," Proceedings, *2006 AIAA Guidance, Navigation, and Control Conference*, Keystone, Colorado, Aug. 21-24, 2006.

[8] John Nash, "Noncooperative games", *Annals of Mathematics*, vol. 54, pp. 286-295, 1951.

[9] J. J. Salerno, , M. Hinman, and D. Boulware, "A Situation Awareness Model Applied To Multiple Domains", In *Proc of the Defense and Security Conference*, Orlando, FL, March 2005.

[10] George Tadda, John Salerno, Douglas Boulware, Michael Hinman and Samuel Gorton, "Realizing Situation Awareness within a Cyber Environment", In *Proceedings of SPIE Vol. 6242* (SPIE, Bellingham, WA, 2006) 624204.

[11] E. Blasch and S. Plano, "DFIG Level 5(User Refinement) issues supporting Situational Awareness Reasoning," Int. *Society of Information Fusion Conference – Fusion05*, 2005.

[12] E. Dijkstra. E. "A Note on Two Problems in Connection with Graphs." *Numerical Mathematics*, October 1959.

[13] W. Stallings, High-Speed Networks: *TCP/IP and ATM Design Principles*, Prentice-Hall, 2002.

[14] D. Shen, G. Chen, J. B. Cruz, Jr., L. Haynes, M. Kruger, and E. Blasch, "A Markov game theoretic approach for cyber situational awareness", accepted for presentation in *SPIE's Defense and Security Symposium*, Orlando, FL, 9-13 April 2007.

[15] A. Purdy, Jr. "Forging a national cyber security strategy", SC Magazine, Mar 2006, http://www.scmagazine.com/asia/news/article/544832/forging-national-cyber-security-strategy/

[16] S. Mathew, D. Britt, R Giomundo, S. Upadhyaya, M. Sudit, and A. Stitz, "Real-time multistage attack awareness through enhanced intrusion alert clustering," *IEEE MILCOM*, 2005

[17] S. J . Yang, J. Holsopple, and M Sudit, "Evaluating Threat Assessment for Multi-Stage Cyber Attacks", *IEEE MILCOM*, 2006.

[18] B. D'Ambrosio, M. Takikawa, J. Fitzgerald, D. Upper, and S. Mahoney, "Security Situation Assessment and Response Evaluation (SSARE), DARPA Information Survivability Conference & Exposition II, 2001. DISCEX '01.Proceedings, Volume 1, 12-14 June 2001 Page(s):387 - 394 vol.1